**A. Monopoly. Key Concepts:**

* Single seller with market power
* Price maker, not price taker
* Sets price where MR = MC
* Always charges p > MC (markup pricing)

**Monopoly Optimization:**

1. Find inverse demand: p(Q)
2. Calculate revenue: R(Q) = p(Q)Q
3. Calculate marginal revenue: MR = dR/dQ
4. Set MR = MC to find optimal Q
5. Substitute Q in demand to find optimal p

**Monopoly Markup Rule:**

* p = MC × (ε/(ε+1))
* Where ε = price elasticity of demand (negative number)

**Deadweight Loss (DWL):**

* DWL = (p^M - MC)(q^PC - q^M)/2
* Represents efficiency loss from monopoly pricing

**Common Calculations:**

* For linear demand: p = a - bQ
* MR = a - 2bQ
* If MC = c, then Q\* = (a-c)/(2b)
* p\* = (a+c)/2
* Monopoly profit = (p\*-c)Q\* = (a-c)²/(4b)

**B. Perfect Competition Key Concepts:**

* Many small firms, price takers
* Homogeneous products
* Free entry/exit
* Perfect information
* Zero economic profits in long run

**Equilibrium Conditions:**

* Short run: p = MC for each firm
* Long run: p = MC = min(AC)

**Market Supply:**

* Short run: Horizontal sum of firms' MC curves above AVC
* Long run: Horizontal sum of firms' MC curves at min(AC)

**Market Efficiency:**

* Allocatively efficient: p = MC
* Productively efficient: production at min(AC)
* No deadweight loss

**Key Comparisons to Monopoly:**

* p^PC < p^M
* q^PC > q^M
* π^PC = 0 < π^M
* DWL^PC = 0 < DWL^M

**C. Oligopoly Models. 1. Cournot Competition Key Features:**

* Firms choose quantities simultaneously
* Strategic substitutes (if rival increases Q, firm decreases Q)
* Nash equilibrium where each firm's quantity maximizes profit given rivals' quantities

**Solving Cournot (n firms with identical MC = c):**

1. Find inverse demand: p(Q) where Q = Σq\_i
2. Firm i profit: π\_i = p(Q)q\_i - cq\_i
3. FOC: ∂π\_i/∂q\_i = 0
4. Find best response function: q\_i(q\_-i)
5. Symmetric equilibrium: q\_i = q\_j for all i,j
6. For linear demand p = a-bQ:
   * q\_i^C = (a-c)/[b(n+1)]
   * Q^C = n(a-c)/[b(n+1)]
   * p^C = a - bQ^C = c + (a-c)/(n+1)
   * π\_i^C = b(q\_i^C)²

**Cournot with Different Marginal Costs:**

* Firm with lower MC produces higher quantity
* Total market quantity increases with number of firms
* As n→∞, p→MC (perfect competition result)

**2. Bertrand Competition Key Features:**

* Firms choose prices simultaneously
* Strategic complements (if rival increases p, firm increases p)
* Homogeneous products: all consumers buy from lowest-price firm

**Bertrand Equilibrium (Homogeneous Products):**

* With identical MC = c: p^B = c (price equals marginal cost)
* Profits: π^B = 0 (competitive outcome)
* With different marginal costs: p^B = min{c₂, monopoly price} where c₂ is second-lowest MC

**Bertrand with Product Differentiation:**

* Linear demand: q\_i = a - bp\_i + dp\_j where d < b
* Best response: p\_i(p\_j) = (a + dp\_j + bc)/(2b)
* Nash equilibrium: p\_i\* = [a(2b+d) + bc(2b+d)]/(4b²-d²)

**3. Stackelberg Model Key Features:**

* Sequential quantity choice (leader moves first)
* First-mover advantage
* Solved using backward induction

**Solving Stackelberg (Linear Demand):**

1. Find follower's best response: q\_F(q\_L) = (a-c-bq\_L)/(2b)
2. Leader substitutes follower's BR: π\_L = (a-b(q\_L+q\_F(q\_L)))q\_L - cq\_L
3. Leader's optimal quantity: q\_L^S = (a-c)/(2b)
4. Follower's quantity: q\_F^S = (a-c)/(4b)
5. Total output: Q^S = (3/4)((a-c)/b)
6. Profits: π\_L^S > π\_F^S

**Comparison across models (same MC = c, linear demand p = a-bQ):**

* Total output: Q^PC > Q^S > Q^C > Q^M
* Price: p^M > p^C > p^S > p^PC
* Profits: π^M > π\_L^S > π\_i^C > π\_F^S > π^PC

II. Game Theory **Key Concepts:**

* Players, strategies, payoffs
* Nash Equilibrium: no player has incentive to deviate unilaterally
* Dominant strategy: best regardless of what others do
* Dominated strategy: always worse than another strategy

**Finding Nash Equilibria:**

1. Identify strictly dominated strategies and eliminate
2. Find best responses for each player
3. Nash equilibria are strategy profiles where all strategies are best responses

**Pure vs. Mixed Strategies:**

* Pure: deterministic choice of action
* Mixed: probability distribution over actions

**Solving Mixed Strategy Equilibria (2×2 Games):**

1. Player must be indifferent between pure strategies in support
2. For player 1 mixing (U,D) with probabilities (p,1-p):
   * EU(player 2 plays L) = EU(player 2 plays R)
   * Solve for p
3. For player 2 mixing (L,R) with probabilities (q,1-q):
   * EU(player 1 plays U) = EU(player 1 plays D)
   * Solve for q

**Coordination Games:**

* Multiple Nash equilibria
* May have Pareto-dominant equilibrium

**Prisoner's Dilemma:**

* Dominant strategy: defect/non-cooperative action
* Socially inefficient outcome

**B. Dynamic Games Key Concepts:**

* Sequential moves
* Information sets
* Subgame Perfect Nash Equilibrium (SPNE)
* Credible vs. non-credible threats

**Solving Dynamic Games:**

1. Identify subgames (starting from terminal nodes)
2. Find Nash equilibria in each subgame
3. Work backwards to find SPNE

**Extensive Form vs. Normal Form:**

* Extensive: tree diagram showing sequence of moves
* Normal: payoff matrix showing all possible strategy combinations

**Pure Strategy in Dynamic Games:**

* Complete contingent plan: what to do in every information set

**Common Dynamic Games:**

* Entry deterrence
* Sequential bargaining
* Ultimatum game
* Centipede game

**C. Repeated Games. Key Concepts:**

* Stage game repeated over time
* Discount factor δ (0 < δ < 1)
* Grim trigger strategy
* Folk theorem: cooperation possible with patient players

**Infinitely Repeated Games:**

* Payoff: π = (1-δ)[π₁ + δπ₂ + δ²π₃ + ...]
* For constant stream: π = π₁/(1-δ)

**Cooperation Condition:**

* Short-term gain from deviation < long-term loss from punishment
* For Prisoner's Dilemma: δ ≥ (π^D - π^C)/(π^D - π^N)
  + π^D: deviation payoff
  + π^C: cooperation payoff
  + π^N: Nash equilibrium payoff

**Strategies in Repeated Games:**

1. Grim trigger: cooperate until deviation, then punish forever
2. Tit-for-tat: copy opponent's previous move
3. Forgiving strategies: return to cooperation after punishment period

**Factors Affecting Cooperation:**

* Higher δ → easier to sustain cooperation
* Higher monitoring ability → easier to detect deviation
* Stronger punishment → easier to sustain cooperation

**A. Expected Utility Theory. Key Concepts:**

* Expected utility: EU = Σp\_i·u(W\_i)
* Risk aversion: u'(W) > 0, u''(W) < 0
* Risk neutrality: u(W) = W or u''(W) = 0
* Risk loving: u''(W) > 0

**Common Utility Functions:**

* Log utility: u(W) = ln(W)
* Power utility: u(W) = W^α where α < 1 (risk averse)
* Exponential utility: u(W) = -e^(-rW) where r > 0 (constant absolute risk aversion)

**Risk Premium and Certainty Equivalent:**

* Risk premium (RP): expected value - certainty equivalent
* Certainty equivalent (CE): u(CE) = EU(lottery)
* For small risks, RP ≈ ½·σ²·(-u''(W)/u'(W))
* Arrow-Pratt measure of risk aversion: -u''(W)/u'(W)

**Optimal Investment Problems:**

1. Find expected utility: EU = Σp\_i·u(W₀ + r\_i·x)
2. Maximize EU with respect to x (investment amount)
3. FOC: ∂EU/∂x = 0
4. Solve for optimal x\*

**B. Insurance Markets. Full Insurance:**

* Eliminates all risk
* For actuarially fair premium, risk-averse individuals always choose full insurance
* Premium = expected loss = Σp\_i·L\_i

**Partial Insurance:**

* Only covers portion of potential loss
* Optimal if premium has loading factor (premium > expected loss)

**Insurance Market Problems:**

1. Calculate expected utility with insurance: EU\_ins = u(W₀ - P)
2. Calculate expected utility without insurance: EU\_no = Σp\_i·u(W₀ - L\_i)
3. Individual buys insurance if EU\_ins ≥ EU\_no

**A. Adverse Selection Key Concepts:**

* Hidden type problem
* Agents know their type, principal doesn't
* Market failure: "lemons problem"
* High-risk/low-quality types drive out low-risk/high-quality types

**Adverse Selection in Insurance:**

* Good risks (low probability of loss) vs. bad risks (high probability)
* With one pooling premium, good risks may leave market
* Equilibrium premium: P = Σ(p\_i·L·f\_i)/Σf\_i where f\_i is fraction of type i who buy insurance

**Solutions to Adverse Selection:**

1. Screening: principal designs mechanism to separate types
   * Menu of contracts with different prices/coverage
   * Self-selection constraint: each type prefers contract designed for them
2. Signaling: informed agents take costly action to reveal type
3. Mandatory participation: forces all types to participate

**B. Moral Hazard. Key Concepts:**

* Hidden action problem
* Agent's unobservable effort affects outcomes
* Principal can observe outcome but not effort

**Principal-Agent Model:**

1. Agent's utility: u(w) - c(e) where w is wage, e is effort
2. Principal's profit: f(e) - w where f(e) is output
3. Participation constraint (PC): E[u(w)] - c(e) ≥ Ū (reservation utility)
4. Incentive compatibility constraint (ICC): e\* = argmax{E[u(w)] - c(e)}

**First-Best Solution (Observable Effort):**

* Set w to satisfy PC with equality
* Agent chooses efficient effort level

**Second-Best Solution (Unobservable Effort):**

* Performance-based pay (incentive contract)
* Trade-off between optimal risk-sharing and incentives
* High-powered incentives increase effort but impose risk on agent

**Moral Hazard in Insurance:**

* Full insurance reduces incentive for precautionary effort
* Solutions: deductibles, co-payments, experience rating

**C. Contracts and Mechanism Design Second-Degree Price Discrimination:**

* Cannot directly observe consumer types
* Offer menu of options (quality-price pairs)
* Self-selection through incentive compatibility

**Solving for Optimal Contracts:**

1. Identify incentive compatibility constraints (ICC)
   * Type H prefers H contract: u\_H(q\_H,p\_H) ≥ u\_H(q\_L,p\_L)
   * Type L prefers L contract: u\_L(q\_L,p\_L) ≥ u\_L(q\_H,p\_H)
2. Identify participation constraints (PC)
   * Type H: u\_H(q\_H,p\_H) ≥ 0
   * Type L: u\_L(q\_L,p\_L) ≥ 0
3. Maximize profit subject to constraints
4. Results:
   * No distortion at the top: high type gets efficient quality
   * Distortion at the bottom: low type gets inefficiently low quality
   * High type's ICC binds, low type's PC binds

**1. 3deg Price Discrimination Steps:**

1. Identify separate markets with different demand elasticities
2. For each market i, find marginal revenue: MR\_i = ∂(p\_i·q\_i)/∂q\_i
3. Set MR\_i = MC for each market
4. Calculate price in each market using demand function
5. Verify p\_i/p\_j = [(ε\_i+1)/(ε\_j+1)]·[ε\_j/ε\_i] where ε is elasticity

**For Linear Demand:**

* If p\_i = a\_i - b\_i·q\_i, then MR\_i = a\_i - 2b\_i·q\_i
* Optimal q\_i\* = (a\_i - MC)/(2b\_i)
* Optimal p\_i\* = (a\_i + MC)/2

**For Isoelastic Demand:**

* If q\_i = D\_i·p\_i^σ\_i where σ\_i < -1
* Optimal p\_i\* = MC·[σ\_i/(σ\_i+1)]
* Price ratio: p\_i/p\_j = [(σ\_i+1)/(σ\_j+1)]·[(σ\_j)/(σ\_i)]

**2. 2deg Price Discrimination Steps for Two Types:**

1. Identify each type's WTP for different versions
2. Set up incentive compatibility constraints
   * High type: WTP\_H(high) - p\_high ≥ WTP\_H(low) - p\_low
   * Low type: WTP\_L(low) - p\_low ≥ WTP\_L(high) - p\_high
3. Set up participation constraints
   * High type: WTP\_H(high) - p\_high ≥ 0
   * Low type: WTP\_L(low) - p\_low ≥ 0
4. Maximize profit: π = f\_H·(p\_high - c\_high) + f\_L·(p\_low - c\_low)
   * f\_H: fraction of high types
   * f\_L: fraction of low types

**Optimal Pricing:**

* p\_low = WTP\_L(low)
* p\_high = WTP\_H(high) - [WTP\_H(low) - WTP\_L(low)]

**Alternative Strategies:**

1. Sell high version to high types, low to low types
2. Sell only high version to high types
3. Sell only low version to both types
4. Sell high version to both types  
   → Compare profits under each strategy

**B. Game Theory Problem**

**Steps for Pure Strategy NE:**

1. For each player i and each strategy s\_i:
   * Find best response to all possible strategies of other players
   * Mark cells where strategy is best response
2. Nash equilibria are cells marked for all players

**Steps for Mixed Strategy NE (2×2 games):**

1. Calculate expected utility for each pure strategy given opponent's mixed strategy
2. Set expected utilities equal and solve for mixing probabilities
   * For row player mixing (p,1-p): p·[EU(L)-EU(R)] = EU(R)
   * For column player mixing (q,1-q): q·[EU(U)-EU(D)] = EU(D)

**Dominated Strategies:**

* Strategy s\_i is strictly dominated if ∃s\_i' such that u\_i(s\_i',s\_-i) > u\_i(s\_i,s\_-i) for all s\_-i
* Iteratively eliminate strictly dominated strategies

**2. Dynamic Games Steps for SPNE:**

1. Draw extensive form (game tree)
2. Identify all subgames
3. Starting from terminal nodes:
   * Find optimal action for player at each decision node
   * Replace subgame with its payoff
4. Continue working backwards until reaching initial node

**3. Repeated Games**

**Steps for Evaluating Cooperation:**

1. Calculate one-shot deviation payoff (π^D)
2. Calculate cooperation payoff (π^C)
3. Calculate Nash equilibrium payoff (π^N)
4. Find minimum discount factor: δ ≥ (π^D - π^C)/(π^D - π^N)

**For Collusion in Cournot:**

* π^C = π^M/n (share of monopoly profit)
* π^D = profit from best response to q\_j = q^M/n
* π^N = Cournot equilibrium profit

**C. Decision Uncertainty**

**Steps for Optimal Investment:**

1. Find expected utility: EU = Σp\_i·u(W₀ + r\_i·x)
2. Maximize EU with respect to x
3. FOC: Σp\_i·u'(W₀ + r\_i·x)·r\_i = 0
4. Solve for optimal x\*

**Insurance Problems:**

1. Calculate expected utility with insurance: EU\_ins = u(W₀ - P)
2. Calculate expected utility without insurance: EU\_no = Σp\_i·u(W₀ - L\_i·I{loss occurs})
3. Calculate certainty equivalent: u(CE) = EU\_no
4. Maximum willingness to pay: P\_max = W₀ - CE

**Risk Premium Calculation:**

* RP = E[W] - CE
* For small risks: RP ≈ ½·Var(W)·(-u''(W̄)/u'(W̄))